Predictors Based on Distributed ARTMAP Neural Network

Anatoli Nachev
Information Systems,
Dept. of Accountancy & Finance,
National University of Ireland,
Galway, Ireland

Seamus Hill
Information Systems,
Dept. of Accountancy & Finance,
National University of Ireland,
Galway, Ireland

Teodosi Teodosiev
Dept. of Computer Science,
Shumen University,
Shumen, Bulgaria

Abstract *- An important task for a direct mailing company is to detect potential customers in order to avoid unnecessary and unwanted mailing. This paper describes a non-linear methodology to predict profiles of potential customers using dARTMAP neural networks. The paper discusses advantages of the proposed approach over similar techniques based on MLP neural networks.*

Keywords: neural networks, adaptive resonance theory, dARTMAP, data mining.

1.0 Introduction

Direct mailings to a company's potential customers, or 'junk mail' to many can be a very effective way for to market a product or service. However, much of this junk mail is really of no interest to the majority of people that receive it.

The task how to predict the profiles of potential customers for a product, given information about the clients and a test sample of customers possessing the particular product is a well-known data mining problem from the world of direct marketing.

Traditionally direct marketing companies have used statistical techniques such as linear regression, decision trees and MLP neural networks to predict which customers are likely to respond or purchase the product.

This paper proposes a non-linear approach based on dARTMAP neural networks to solve this task.

Section 1 outlines the prediction task, a variety of approaches used to solve it.

Section 2 discusses the main characteristics of a predictor based on the dARTMAP model and outlines its algorithm.

Section 3 describes the preprocessing steps needed to prepare a dataset in order to be used by a dARTMAP network.

Section 4 describes experiments conducted with a dARTMAP neural network simulator, and discusses results.

1.1 Prediction Task

The dataset used to test the proposed approach is based on real world business data [12]. It is a block of very detailed survey information on the people some of whom bought and plan to buy a caravan insurance policy. The people were asked to answer 85 questions, each of which can be regarded as one feature in the classification. The block consists of 3 parts. The first is training data, which contains a number of survey responses, some of which come from caravan policy holders. The second part is testing data, and it contains answers from potential caravan insurance policy buyers. The last part is the true data that shows who of those potential buyers actually bought the policy at last.

In the prediction task, the underlying problem is to find the subset of customers with a probability of having a caravan insurance policy above some boundary probability.

A wide variety of methodological approaches were used to solve this prediction task. Methods include: standard statistics [12], backpropagation neural networks [1], [8], [13], self-organizing maps (SOMs) [14], genetic programming, C4.5, CART, and other decision tree induction algorithms, fuzzy clustering and rule discovery, support vector machines (SVMs), logistic regression, boosting and bagging, and more [12]. The best technique for prediction reported in [9] and [12] is the Naive Bayesian learning, provided 800 predictions made, which gives a hit rate about 15.2%. Predictors based on the backpropagation MLP networks show accuracy rate about 71% and hit rate about 13% as reported in [1], [2], [8], and [12].

2.0 dARTMAP vs. MLP

ART is a family of neural networks for fast learning, pattern recognition, and prediction, including both unsupervised: An ART model is designed to guarantee stable memories even with fast on-line learning. However, ART stability typically requires winner-take-all (WTA) coding, which may cause category proliferation in a noisy input environment. While ART code representations may be distributed in theory, in practice nearly all ART networks feature WTA coding [6].

From another hand, a multi-layer perceptron (MLP) employs slow off-line learning to avoid catastrophic forgetting in an open input environment, which limits adaptation for each input and so requires multiple presentations (epochs) of the training set. With fast learning, MLP memories suffer catastrophic forgetting. Features of a fast-learn system, such as its ability to encode significant rare cases and to learn quickly in the field, may be essential for the given application domain.

An ART module is embedded as the primary component of ARTMAP, and similarly an unsupervised dART module is embedded in a supervised dARTMAP network [5]. A dART system combines the computational advantages of ART and MLP systems [3]. Properties include code stability when learning is fast and on-line, memory compression when inputs are noisy and unconstrained [7]. The coding field of a dARTMAP supervised system is analogous to the hidden layer of a multi-layer perceptron (MLP), where distributed activation helps the network achieve memory compression and generalization.

2.1 dARTMAP Algorithm

Figure 1 represents simplified dARTMAP architecture [5], [6]. In the general case, dARTMAP learns to predict an arbitrary outcome vector $b = (b_1, \dots, b_k, \dots, b_L)$, given an input vector $a = (a_1, \dots, a_i, \dots, a_M)$. Each dARTa input is complement coded, with $0 \le a_i \le 1$, so $I = A = (a, a^c)$, i.e.

$$A_i = \begin{cases} a_i & if\ 1 \le i \le M \\ 1 - a_i & if\ M+1 \le i \le 2M \end{cases}$$

During dARTMAP training, the input pairs $(a^{(1)}, b^{(1)}), (a^{(2)}, b^{(2)}), \dots, (a^{(n)}, b^{(n)}), \dots$ are presented for equal time intervals.

A complement-coded input A activates a distributed F_2 code y, which in turn is filtered through counting weights c_j to produce the F_3 activation Y. The WTA field F_0^{ab} activates the node $k = K'$ that receives the largest input σ_k from F_3, representing the predicted output class. During training, activation at the field F_1^{ab} determines whether the predicted output class $k = K'$ matches the actual output class $k = K$, which is represented at the field F_0^b. Adaptation in paths from F_0^b to the coding field F_2 realises credit assignment. A mismatch at F_1^{ab} causes a match tracking signal to raise ARTa vigilance ρ just enough to reset the active code.

Each dARTMAP input first activates a distributed code. If this code produces a correct prediction, learning proceeds in the distributed coding mode. If the prediction is incorrect, the network resets the active code via match tracking feedback. In ARTMAP networks, the reset process triggers a search for a category node that can successfully code the current input. It also places the system in a WTA coding mode for the duration of the search. In WTA mode, dARTMAP can, like ARTMAP, add nodes incrementally as needed. When a coding node is added to the network, it becomes permanently associated with the output class that is active at the time. From then on, the network predicts this class whenever the same coding node is chosen in WTA mode.

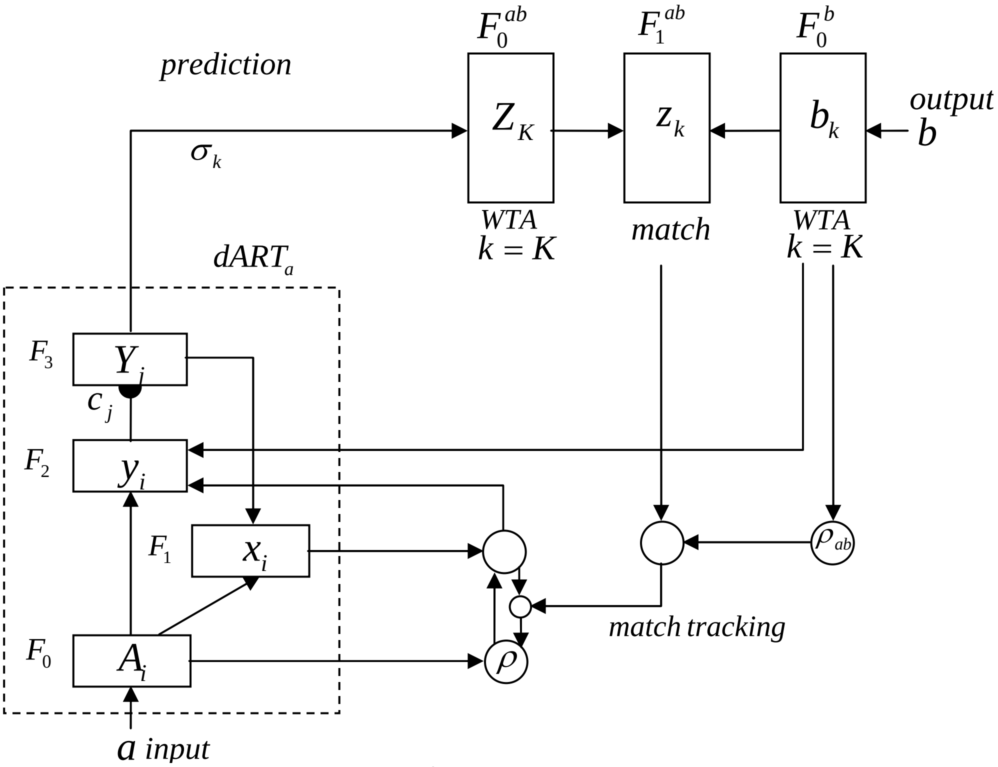

Figure 1. Simplified dARTMAP architecture.

3.0 Data Preprocessing

The dataset used for dARTMAP NN simulations is owned and supplied by the data mining company Sentient Machine Research [12].

The train dataset contains 5822 customer records. Each record consists of 86 attributes containing socio-demographic data represented by attributes 1-43 and product ownership attributes 44-86. The socio-demographic data is derived from zip codes. All customers living in areas with the same zip code have the same socio-demographic attributes. Attribute 86, "CARAVAN: Number of mobile home policies", is the target variable.

Evaluation dataset for validation of the prediction model consists of 4000 customer records. It has the same format as the training dataset, only the target is missing. Targets for the evaluation set have been provided by a separate file.

3.1 Feature selection

Feature selection is very critical for solving the prediction task. Set of input features submitted to the classifier affect its performance, training time, and efficiency of storage.

To increase the chance of identifying people who are likely to hold or possibly be in the market for a caravan insurance policy, we focus on the following characteristics:

1. Car owners with high contribution to car policy purchases. Those who do not have a car are unlikely to own a caravan, as they generally require to be towed. Car owners can be readily identified as those having existing car insurance policies.

The amount spent on policies is also important. People who spend more on car insurance are most likely to be caravan policy buyers, and the more they spend, the more likely a buyer they are.

2. People having fire policy with high level of contribution. This may indicate that the fire insurance is for a caravan. The level of the fire insurance cover that is most likely to be indicative of a caravan policy is level 4.

3. People having a high level of purchasing power. Apart from 'Purchasing Power Class', all socio-demographic attributes, including customer segmentations by lifestyle, income, etc., often do not add any predictive power when behavioral data is available. People with high purchasing power are not necessarily enthusiastic about insuring their property, but they do have quite enough wealth to own a caravan, even if using it were not their prime hobby. Typical customers have high, or at least medium, education, status, social class, and income levels. For the feature selection, all demographic attributes were discarded, except attribute 43, "MKOOPKLA Purchasing power class".

In conclusion, we suggest a partial customer profile based on the available input features:

- Attribute 43 (MKOOPKLA) Purchasing power class
- Attribute 47 (PPERSAUT) Contribution car policies
- Attribute 59 (PBRAND) Contribution fire policies

The choice of these attributes is justified by numerous evaluations of the relative attribute importance and sensitivity for the prediction task, e.g. greedy feature selection algorithm, statistics, stepwise procedures, evolutionary algorithms, chi analysis [12], and others.

Intuitively, these three predictors identify customers who have a car and are wealthier than average, and who in general carry more insurance coverage than average. It is not surprising that these are the people who are most likely to have caravan insurance.

4.0 Benchmarks

The experiments aimed to explore:

- If the dARTMAP model is sensitive to the order in which features and input patters are submitted. This is due to the fact that in some ART models the LTM nodes commitment during the training depends on this order.
- The optimal values of the network parameters, accuracy rate, and hit rate.
- Role of the network parameters in the model performance in terms of train time, test time, and memory.

To maximize use of the datasets and to avoid bias in the selection of the training and test sets, a cross-validation technique was applied. Cross-validation created N copies of the classifier and tested each on 1/N of the evaluation dataset, after training it on 1/N-th of the training set. In other words, each classifier makes predictions for its 1/N-th of the data, yielding predictions for the whole set. Cross-validation was applied using N=5.

Results form the first group of experiments showed that the dARTMAP model is sensitive to the order in which features appear in the feature set, and the order in which input patterns appear. Out of six permutations of the attributes 43, 47, and 59, only three (50%) gave satisfactory level of prediction: {43, 47, 59}, {43, 59, 47}, and {59, 43, 47}. To see how the sequence of input patters affect the predictability, the network was trained with eight different order input sets: initial order; real buyer entries shifted to the beginning and the end, respectively; and five randomly chosen sequences by using views of the dataset. All results show that the extreme cases of buyers shifted to both ends don’t yield acceptable predictive results. The model, however, does not make any difference between the other six sequences. The results reported further were based on the initial order of input patterns.

To see how the network parameters influence the predictiveness, simulations with a full range of parameter values were conducted. Results show that an acceptable level of predictiveness can be achieved by the following values only : $\rho_{test} = 0$, $\alpha = 0.01$, $\beta = 1.0$, $\varepsilon = -0.001$, and $p = 1.0$. The vigilance parameter ρ (Rhobar) was set to various values in order to change the level of details and granularity of the clusters, thus to vary accuracy of predictions, hit rate, and network performance.

Figure 2 shows the values where the vigilance parameter gives highest accuracy, nearly 94%. The best result achieved was 141 predictions made with confusion matrix shown in Table 1. Figure 3 shows how the vigilance affects the hit rate. The dARTMAP model reaches 30% hit rate with parameter values between 0.935 and 0.96. This result exceeds reported 13% of the MLP networks for the same prediction task. Figure 4 and Figure 5 reveal another advantage of the dARTMAP model over MPL networks, namely the real time performance. The vigilance parameter varies slightly the train and test times, about 5 and 0.7 seconds respectively. For the same prediction task an MLP network requires about 35 minutes training [13]. Figure 6 shows the long-term memory in kilobytes required by the model, which in the worst case scenario does not exceed 24K.

		Predicted		
		No	Yes	
Actual	No	3634	128	3762
	Yes	225	13	238
		3859	141	4000

Table 1. Confusion matrix of feature set {59, 43, 47} and vigilance parameter $0.935 \le \rho < 0.96$

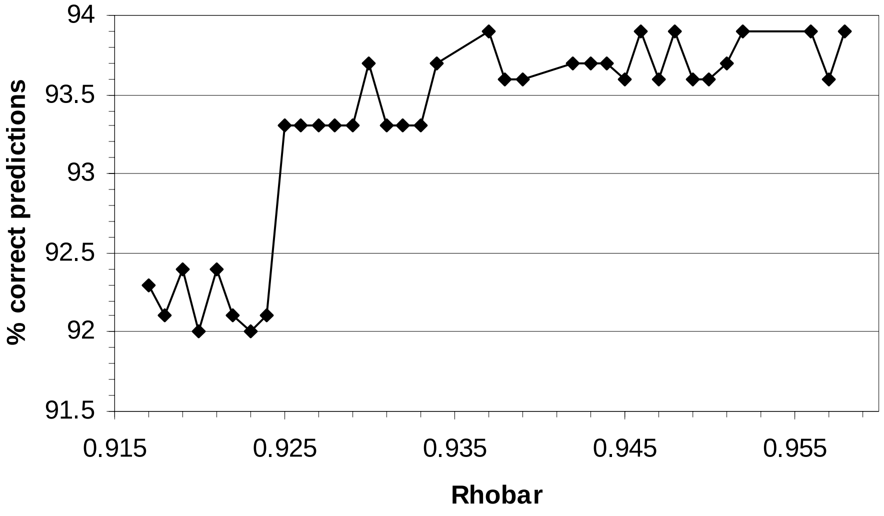

Figure 2. Prediction accuracy of the dARTMAP model with feature set {59, 43, 47} and $0.915 \le \rho < 0.96$.

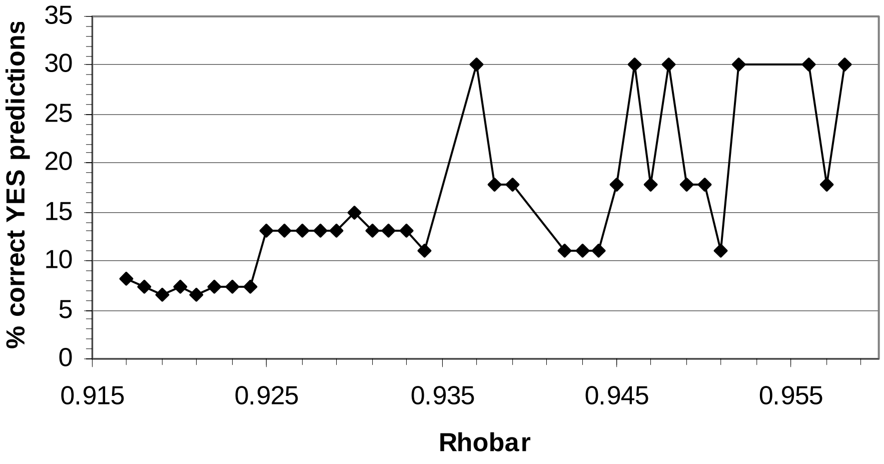

Figure 3. Hit rate of the dARTMAP model with feature set {59, 43, 47} and $0.915 \le \rho < 0.96$.

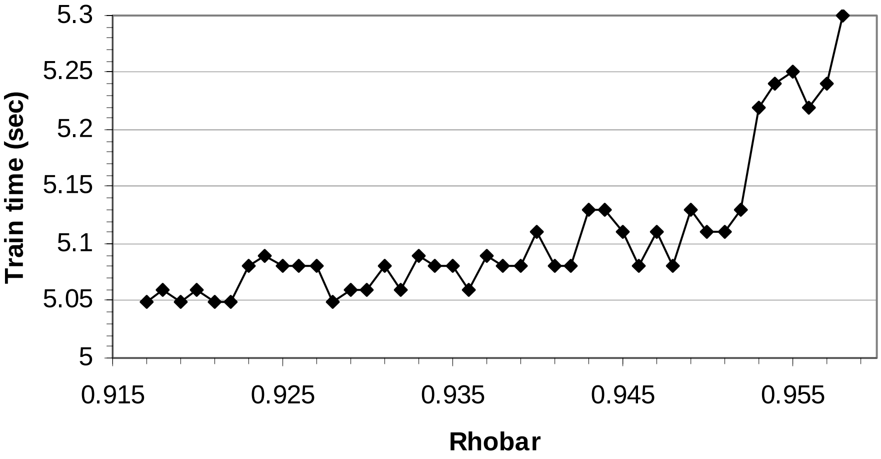

Figure 4. Train time in seconds of the dARTMAP model with feature set {59, 43, 47} and $0.915 \leq \rho < 0.96$.

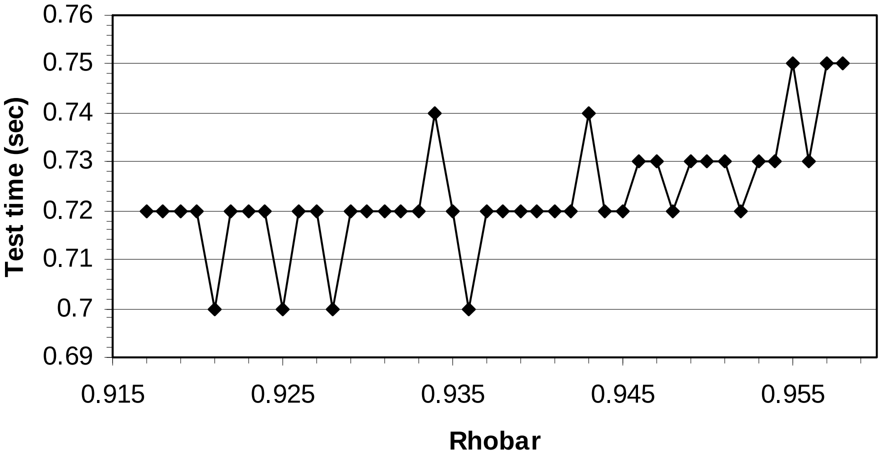

Figure 5. Test time in seconds of the dARTMAP model with feature set {59, 43, 47} and $0.915 \leq \rho < 0.96$.

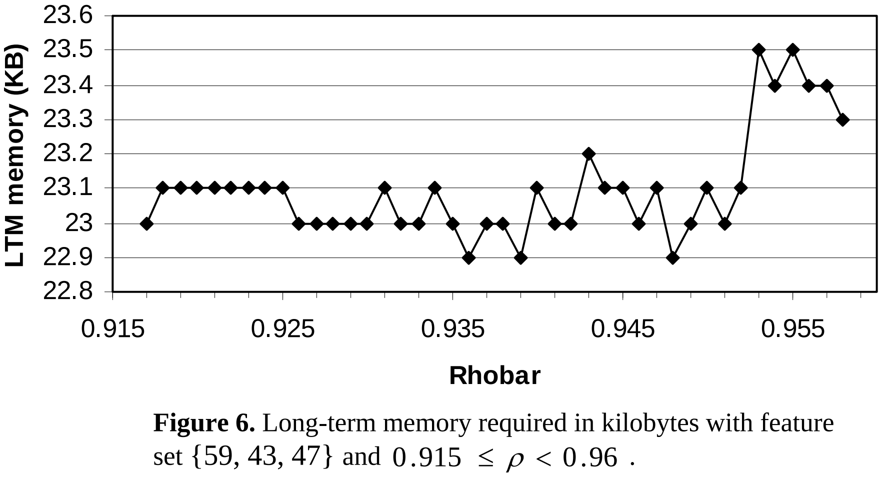

Figure 6. Long-term memory required in kilobytes with feature set {59, 43, 47} and $0.915 \leq \rho < 0.96$.

5.0 Conclusion

This paper proposes a non-linear approach, for solving a prediction task to identify potential buyers of insurance policy. Our approach is based on dARTMAP neural networks, because they combine the computational advantages of MLP and ART systems. The primary benefit of this approach is code stability when learning is fast and on-line.

The simulation results show that a predictor based on a dARTMAP NN outperforms similar techniques based on backpropagation MLP NNs in terms of hit rate, overall accuracy, and train time.

6.0 References

[1] Brierly, P. (2000) Characteristics of caravan insurance policy owners, Available at http://www.liacs.nl/~putten/library/cc2000/brierl~1.pdf

[2] Candocia, F. (2004) EEL 6825 Pattern Recognition, Available at http://www.cise.ufl.edu/~bfeng/eel6825/ 6825_pattern.html

[3] Carpenter, G.A. (1996). Distributed ART networks for learning, recognition, and prediction. Proceedings of the World Congress on Neural Networks (WCNN’96), pp. 333–344.

[4] Carpenter, G.A., Grossberg, S., & Reynolds, J.H. (1991). ARTMAP: Supervised real–time learning and classification of nonstationary data by a self–organizing neural network. Neural Networks, 4, 565–588.

[5] Carpenter, G.A., Milenova, B., & Noeske, B. (1998). dARTMAP: A neural network for fast distributed supervised learning. Neural Networks, 11, 793-813. Technical Report CAS/CNS TR-97-026, Boston, MA: Boston University.

[6] Carpenter, G.A., & Milenova, B.L. (1999). Distributed ARTMAP. Proceedings of the International Joint Conference on Neural Networks (IJCNN'99), Technical Report CAS/CNS TR-99-013, Boston, MA: Boston University.

[7] Carpenter, G.A. (2000). ART neural networks: Distributed coding and ARTMAP applications. In Peter Sincák and Ján Vascák (Eds.), Quo Vadis Computational Intelligence? New Trends and Approaches in Computational Intelligence. In the series Studies in Fuzziness and Soft Computing, New York: Physica-Verlag, 3-12. Technical Report CAS/CNS TR-2000-005, Boston, MA: Boston University.

[8] Crocoll, W. (2000) Artificial Neural Network Portion of Coil Study, Available at http://www.liacs.nl/~putten/library/cc2000/crocol~1.pdf

[9] Elkan, C. (2001) Magical Thinking in Data Mining: Lessons From CoIL Challenge 2000. In Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining (KDD'01), pp. 426-431.

[10] Grossberg, S. (1976) Adaptive pattern classification and universal recoding. II: Feedback, expectation, olfaction, and illusions. Biological Cybernetics, 23, 187–202.

[11] Grossberg S. (1980). How does a brain build a cognitive code? Psychological Review, 87, 1–51.

[12] P. van der Putten and M. van Someren (eds). CoIL Challenge 2000: The Insurance Company Case. Published by Sentient Machine Research, Amsterdam. Also a Leiden Institute of Advanced Computer Science Technical Report 2000-09. June 22, 2000.

[13] Shtovba, S. Mashnitskiy, Y. (2000) The Backpropagation Multilayer Feedforward Neural Network Based Competition Task Solution, Available at http://www.liacs.nl/~putten/library/cc2000/shtob~1.pdf

[14] Vesanto, J. Sinkonen, J. (2000) Submission for the Coil Chalange 2000, Available at http://www.liacs.nl/~putten/library/cc2000/vesant~1.pdf